



太原理工大学  
TAIYUAN UNIVERSITY OF TECHNOLOGY

# 2018 届本科生毕业设计(论文)

## 基于区块链的房地产交易平台

学 号： 2018006797  
姓 名： 宋叮咛  
学 院： 大数据学院  
专 业： 软件工程（信息与大数据工程方向）  
班 级： 软件 1840  
指导教师： 王莉 教授

完成日期： 2022 年 5 月

## 声明及论文使用的授权

本人郑重声明：所呈交的毕业设计（论文）是本人在指导教师的指导下取得的研究成果，毕业设计（论文）写作严格遵循学术规范。除了文中特别加以标注和致谢的地方外，毕业设计（论文）中不包含其他人已经发表或撰写过的研究成果。因本毕业设计（论文）引起的法律结果完全由本人承担。太原理工大学享有本毕业设计（论文）的研究成果。

论文作者签名

年 月 日

本毕业设计（论文）作者和指导教师同意太原理工大学保留使用毕业设计（论文）的规定，即：学校有权保留送交毕业设计（论文）的复印件，允许毕业设计（论文）被查阅和借阅；学校可以上网公布全部内容，可以采用影印、缩印或其他复制手段保存毕业设计（论文）。

论文作者签名：

指导教师签名：

签字日期： 年 月 日

签字日期： 年 月 日

太 原 理 工 大 学

# 毕 业 设 计 （ 论 文 ） 任 务 书

第 1 页

毕业设计（论文）题目：

## 基于区块链的房地产交易系统

毕业设计（论文）要求及原始数据（资料）：

1. 综述国内外基于区块链的房地产交易系统的研究现状；
2. 深入了解基于区块链的房地产交易系统的相关技术；
3. 设计并实现基于区块链的房地产交易系统，训练算法分析与设计的能力；
4. 深入分析基于区块链的房地产交易系统中蕴含的关键技术，学会用程序设计语言实现相关的功能；
5. 训练检索文献资料和利用文献资料的能力；（必须包含）
6. 训练撰写技术文档与学位论文的能力。（必须包含）

**毕业设计（论文）主要内容：**

1. 综述基于区块链的房地产交易系统的应用场景；
2. 了解基于区块链的房地产交易系统的相关技术；
3. 熟悉Python、Solidity等程序设计语言，并会配置相应的开发环境；
4. 设计并配置区块链的测试环境与开发环境；
5. 深入分析项目所使用的区块链分析的关键技术；
6. 熟练掌握基于区块链的房地产交易系统的设计与实现方法；
7. 设计与实现基于区块链的房地产交易系统，并能实现具体的功能，保证程序的稳定运行。

**学生应交出的设计文件（论文）：**

1. 内容完整、层次清晰、叙述流畅、排版规范的毕业设计论文；
2. 包括毕业设计（论文）、源程序等毕业设计电子文档及其它相关材料。

## 主要参考文献(资料):

- [1] Savasere A, Omiecinski E, Navathe S. An efficient algorithm for mining association rules in large databases[C]. Proceedings of the International Conference on Very Large Databases, 1995, pp. 432-444.
- [2] Agrawal R, Srikant R. Fast algorithm for mining association rules[C]. Proceedings of the International Conference on Very Large Databases, 1994, pp. 487-499.
- [3] Jiawei Han, Micheline Kamber, Jian Pei. 数据挖掘: 概念与技术 (第三版) [M]. 机械工业出版社, 2012.
- [4] Jiawei Han, Jian Pei, Yiwen Yin. Mining frequent patterns without candidate generation[C]. Proceedings of the International Conference on Management of Data, 2000, pp. 1-12.
- [5] Inokuchi A, Washio T, Motoda H. An Apriori-based algorithm for mining frequent substructures from graph data[C]. Proceedings of the European Symposium on the Principle of Data Mining and Knowledge Discovery, 2000, pp. 13-23.
- [6] Kuramochi M. Kary. Frequent subgraph discovery[C]. Proceedings of the International Conference on Data Mining, 2001, pp. 313-320.
- [7] Xifeng Yan, Jiawei Han. gSpan: graph-based substructure pattern mining[C]. Proceedings of the International Conference on Data Mining, 2002, pp. 721-724.
- [8] 刘勇, 李建中, 高宏. 从图数据库中挖掘频繁跳跃模式[J], 软件学报, 2010, 21(10): 2477-2493.
- [9] 邹兆年, 李建中, 高宏. 从不确定图中挖掘频繁子图模式[J], 软件学报, 2009, 20(11): 2965-2976.
- [10] Xifeng Yan, Jiawei Han. CloseGraph: Mining Closed Frequent Graph Patterns[C]. Proceeding of the International Conference on Knowledge Discovery and Data Mining, 2003, pp. 286-295.

专业班级 \_\_\_\_\_ 软件 1840 班 \_\_\_\_\_ 学生 \_\_\_\_\_ 宋叮咛 \_\_\_\_\_

要求设计(论文)工作起止日期 \_\_\_\_\_ 22 年 01 月 15 日~22 年 06 月 20 日 \_\_\_\_\_

指导教师签字 \_\_\_\_\_ 日期 \_\_\_\_\_

教研室主任审查签字 \_\_\_\_\_ 日期 \_\_\_\_\_

系主任批准签字 \_\_\_\_\_ 日期 \_\_\_\_\_

# 基于区块链的房地产交易平台

## 摘 要

房地产市场在当今社会经济中扮演了越来越重要的作用。同时，房地产市场几乎是最复杂的市场领域之一。房地产交易中，不仅需要高昂的资金投入，还需要繁杂的手续，和各种中介机构的参与。以上的种种原因，不仅提高了欺诈发生的可能性，提高了房地产市场的参与门槛，还降低了房地产市场的流动性。本文讨论并探索了房地产代币化及基于代币的投资方式，并基于房地产代币化的理论，通过以太坊和 Django 实现了房地产交易平台。该平台通过发行房地产代币，通过降低房地产市场的参与门槛，提高房地产交易的安全性，和简化房地产交易的流程等方式，使其便于销售、转让和抵押。同时，代币的引入可以让投资者共享一处房地产的收益，让房地产行业真正普惠广大民众。

**关键词：** 房地产；区块链；代币化

# Blockchain-based Real Estate Trading Platform

## Abstract

The real estate market plays an increasingly important role in today's society and economy. At the same time, the real estate market is almost one of the most complicated market areas. In real estate transactions, not only high capital investment is required, but also complicated procedures and participation of various intermediaries. The above reasons not only increase the possibility of fraud, increase the threshold for participation in the real estate market, but also reduce the liquidity of the real estate market. This article discusses and explores real estate tokenization and tokens -based investment methods, and based on the theory of real estate tokenization, realized real estate trading platforms through Ethereum and Django. By issuing real estate tokens, the platform has made it easy for sales, transfer and mortgage by reducing the entry threshold of the real estate market, improving the security of real estate transactions, and simplifying the process of simplify real estate transactions. At the same time, the introduction of tokens allows investors to share the income of a real estate and allow the real estate industry to truly include the public.

**Key words:** real estate; blockchain; tokenization

# 目录

摘 要 .....	1
Abstract .....	2
目录 .....	1
1. 绪论 .....	1
1.1. 课题背景 .....	1
1.2. 国内外研究现状 .....	1
1.3. 本文研究内容 .....	1
1.4. 本文结构 .....	1
2. 房地产代币化的意义 .....	2
2.1. 消除房地产中介 .....	2
2.2. 增加房地产市场流动性 .....	2
2.3. 增强安全性 .....	2
3. 项目介绍 .....	2
3.1. 功能介绍 .....	3
3.1.1. 注册房地产 .....	3
3.1.2. 租赁 .....	3
3.1.3. 租金设置规则 .....	4
3.1.4. 租金投票权重计算规则 .....	4
3.1.5. 代币转移过程解读 .....	4
3.1.6. 销售、拍卖和转账 .....	5
3.1.7. 架构概览 .....	5
3.1.8. 存储部分 .....	5
3.1.9. 逻辑部分 .....	6
3.1.10. 前端部分 .....	6
3.2. 架构解读 .....	6
3.2.1. 存储部分 .....	6
3.2.2. 逻辑部分 .....	7
3.3. 技术选型 .....	8



3.3.1. 区块链平台 .....	8
3.3.2. 智能合约协议 .....	9
3.3.3. 中心化部分的后端框架 .....	9
3.3.4. Django Restful framework .....	10
3.3.5. 中心化部分的数据库 .....	10
3.3.6. 对象存储服务 .....	10
3.3.7. 前端框架 .....	11
4. 项目实施 .....	11
4.1. 构建智能合约 .....	11
4.2. 实现异步的注册房地产功能 .....	11
4.3. 实现监听智能合约运行状态 .....	12
4.4. 实现前端 .....	13
结论 .....	13
参考文献 .....	13
致谢 .....	13

# 1. 绪论

## 1.1. 课题背景

近年来，我国经济增长面临一定的下行压力，原因之一是房地产市场出现流动性困难的问题。虽然过去几年，房地产市场持续火热，但该市场也几乎是最复杂的投资市场之一。其一，房地产的高价值性、整体性和不可分割性使得房地产天然具有较高的投资门槛。其二，房地产交易过程中具有复杂的流程和条款——尤其是跨国投资，需要遵循当地法律法规和投资环境。这迫使投资者需要通过中介或律师提供投资建议，而中介和律师往往需要收取高额的手续费或咨询费。以上的种种问题限制了房地产市场的活性，不利于投资方式的多元化。[1]

## 1.2. 国内外研究现状

与此同时，区块链相关的技术在近些年得到了快速发展。区块链相关的技术在很大程度上缓解了以往技术存在的成本高、易伪造、不稳定等缺陷。

区块链可以简单地看作一种只增的分布式数据库。它通过共识机制，将数据存储在位于不同国家、不同机构的节点中。并且，使用哈希算法和共识机制，可以保证各个节点的链上数据不被篡改。

智能合约是一种旨在以信息化方式传播、验证或执行合同的计算机协议。智能合约可以帮助参与者在不可信的环境下自动应用合约内容，具有可追踪和不可回滚的性质。智能合约可以部署在区块链上。由于区块链的特性，智能合约一旦部署便不可修改。这显著增加了智能合约的可信程度，使得智能合约的参与方可以在不可信的环境下安全地交易。

代币化，就是将区块链上的数字资产或数字权限使用数字货币表示，是区块链的重要应用之一。将资产代币化，不仅可以省去中介、简化交易流程、加快交易速度，还可以与智能合约结合，降低合约的管理成本、解决交易参与方的不可信问题。

## 1.3. 本文研究内容

本文讨论了房地产代币化的意义，并以此为依据，基于以太坊智能合约和 Django 框架实现了去中心化的房地产交易应用。主要实现了房地产代币的发行、销售、转账和房地产租赁功能。

## 1.4. 本文结构

本文首先讨论了房地产代币化的意义。之后介绍了以此为理论依据实现的去

中心化房地产交易平台的功能、架构、技术选型和实现方法。最后，对本文的工作做出了总结。

## 2. 房地产代币化的意义

使用区块链技术将房地产代币化，可能将缓解当前房地产市场存在的诸多问题。

### 2.1. 消除房地产中介

律师或银行一直是房地产交易过程中的重要部分。然而，使用区块链的在线房地产交易平台可以将中介机构从房地产交易流程中剔除，从而降低成本，使双方受益。

### 2.2. 增加房地产市场流动性

长期以来，房地产一直被认为是一种非流动性资产，因为销售需要很长时间才能结束。加密货币和代币却并非如此，因为从理论上讲，交易双方可以很容易地通过交易所交易房地产代币。卖方可以部分出售他们的房地产代币，从而可以立刻从他们的房产中获得一些价值。

### 2.3. 增强安全性

区块链作为一种去中心化的技术，可以提高可信度和安全性。存储在区块链中的信息可供网络上的所有对等方访问，从而使数据透明且不可变。若回看 2008 年的房地产泡沫崩盘，就能看到贪婪缺乏透明度会产生多么巨大的灾难性的后果。去中心化交易所自身就拥有极高的透明性。由于信息可以很便捷地进行验证，因此买家和卖家在进行交易时可以更有安全感。欺诈时间也将减少。随着法律的日趋完善，智能合约正日益成为可接受的交易方式。因此，智能合约在房地产行业的发展将很有前景。[2]

## 3. 项目介绍

基于区块链的房地产交易平台，是一款去中心化应用。项目试图打破传统房地产的整体性与不可分割性、进一步简化房地产交易流程、并进一步降低房地产交易产生的额外费用。项目试图允许投资者以低投入进入房地产市场，让更多人从房地产市场中获利。本项目发行房地产代币。投资者可以持有部分房地产代币，既可以通过销售或拍卖的方式获利，也可以通过领取房地产租赁分红的方式获利。

这种基于代币的房地产投资方式只能建立在区块链的基础之上。因为区块链

上的数据拥有不可修改和不可删除的特性。运行在区块链上的智能合约，可以在没有第三方的条件下，使得交易各方完成可信的和不可逆转的交易。智能合约可以保证交易的各个环节按照既定流程执行，从而维护交易各方的利益。

本项目属于去中心化应用，与传统的中心化应用有着一定的区别。项目最大的优势在于，其拥有极高的稳定性。因为项目运行于分布在世界各地的以太坊节点，因此不存在单点故障，也拥有良好的抗审查机制——本项目运行于分布在世界各地的以太坊节点上，难以被组织或个人屏蔽。

### 3.1. 功能介绍

#### 3.1.1. 注册房地产

房地产持有者可以在平台中注册已拥有的房地产。注册后，该处房产将由智能合约代理。智能合约将发行该处房产的代币，并将其全部转入注册者账户。此时，注册者可以自由处置持有的代币，例如：将持有的代币转账给亲友，让亲友与自己共享收益；也可以将代币放在交易所中，出售持有的代币。

#### 3.1.2. 租赁

代币持有者获取收益的方式之一是收取租金的分红。每个投资者都可以为房地产设置自己理想的租金，这些租金将按照投资者持有的代币数量为依据进行加权平均，加权平均的租金作为该处房地产的最终租金。

租客可以自定义租赁时长，在缴纳足额的租金后，智能合约会将当前房地产的状态设置为已租赁，并将按照投资者持有的代币比例，为每位投资者分红。

分红时，智能合约将触发 `income` 事件，便于 Celery 周期性任务更新收益信息。

该部分的实现如代码 3.1.2-1 所示。

```
1. // Distribute rent to holders
2. address holder;
3. uint256 votingTokens;
4. uint256 balance = _rentInfos[id].rent * (duration / 86400);
5.
6. for (uint256 i = 0; i < _rentInfos[id].holders.length(); i++) {
7.     holder = _rentInfos[id].holders.at(i);
8.     votingTokens = balanceOf(holder, id);
9.
10.    payable(holder).transfer(balance * votingTokens / 10**18);
11.
12.    emit Income(holder, id, balance * votingTokens / 10**18);
```

代码 3.1.2-1 按持有代币比例分红并触发 `income` 事件

### 3.1.3. 租金设置规则

为了维护市场稳定，也为了维护投资者的共同利益，租金的设置需要遵循一定的规范。

房地产被某人 100% 持有

由于此时持有者 100% 持有该处房地产，因此租金的设置没有任何限制。

房地产被数名（大于等于 2 名）投资者共同持有，且当前投资者之前没有对租金投票

租金必须介于当前租金的 80% 到 120% 之间。

房地产被数名（大于等于 2 名）投资者共同持有，且当前投资者之前已经对租金投票

租金必须介于去除该名投资者后的租金的 80% 到 120% 之间。

### 3.1.4. 租金投票权重计算规则

每位投资者对租金投票时的权重为：投票者持有代币数量占全部已投票投资者持有代币数量的比重。例如，某房地产代币发行量为  $1e18$  枚，A 持有  $1e17$  枚，B 持有  $1e17$  枚，C 持有  $8e17$  枚。A 和 B 均对租金进行了投票，C 没有进行投票，则 A 和 B 的权重各 50%。

### 3.1.5. 代币转移过程解读

房地产代币的发行、转账、交易和销毁本质上都是代币的转移。代币发行时，从 `0x00` 账户转移到投资者账户；代币的转账和交易时，由 A 账户转移到 B 账户；代币销毁时，从投资者账户转移到 `0x00` 账户。由于租金的计算以来投资者持有的代币份额，因此代币的转移也会影响到租金的计算。综上，在代币转移过程中，我们需要对租金重新计算。

智能合约首先获取转账发起者和接收者对房地产租金投票的金额，若双方投票的金额不同，则重新计算加权租金。基本分为三种情况：接收者没有投票、发起者没有投票和双方均投票，单金额不同。

若接收者没有投票。加权租金计算方式为公式 3.1.5-2。

$$voting_{new} = voting_{old} - voting_{transfer}$$

公式 3.1.5-1 接收者没有投票时参与投票的代币更新公式

$$rent_{new} = (rent_{old} - rent_{from} * \frac{voting_{transfer}}{voting_{old}}) * \frac{voting_{old}}{voting_{new}}$$

公式 3.1.5-2 接收者没有投票时，加权租金更新公式

若发起者没有投票。加权租金计算方式为公式 3.1.5-4。

$$voting_{new} = voting_{old} + voting_{transfer}$$

公式 3.1.5-3 发起者没有投票时，参与投票的代币更新公式

$$rent_{new} = rent_{old} * \frac{voting_{old}}{voting_{new}} + rent_{to} * \frac{voting_{transfer}}{voting_{new}}$$

公式 3.1.5-4 发起者没有投票时加权租金更新公式

若双方均投票，但投票金额不同。加权租金计算公式为公式 3.1.5-5。

$$rent_{new} = rent_{old} + (rent_{to} - rent_{from}) * \frac{voting_{transfer}}{voting}$$

公式 3.1.5-5 发起者和接收者均投票，但金额不同时，加权租金更新公式

### 3.1.6. 销售、拍卖和转账

这些操作需要在以太坊代币交易所，如 OpenSea 中进行，这里不做详细介绍。

### 3.1.7. 架构概览

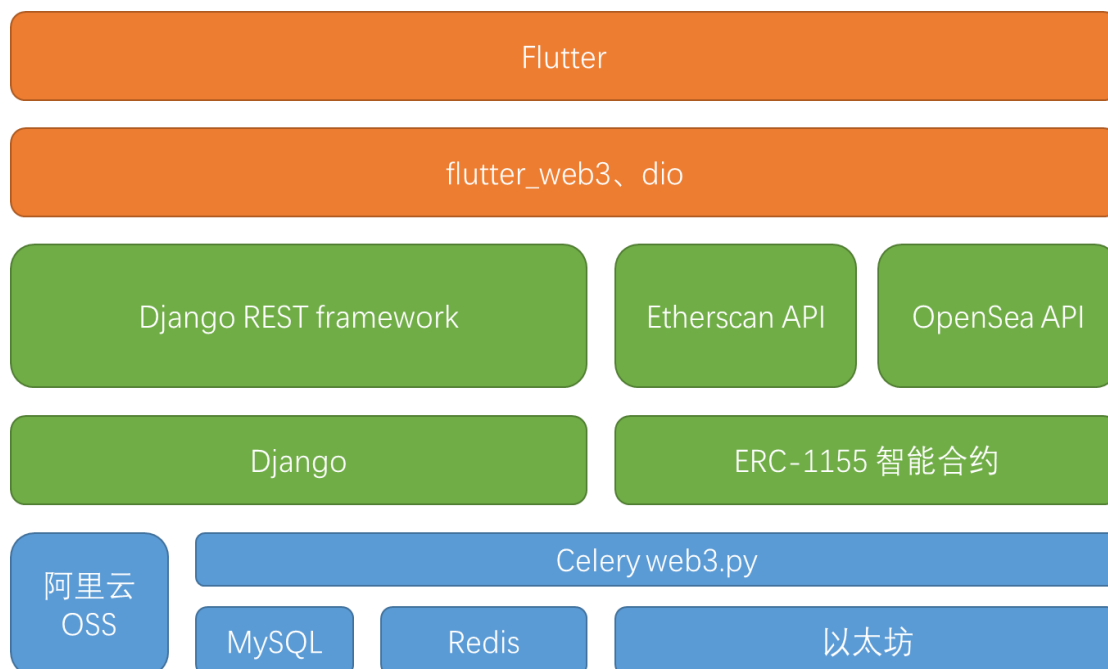


图 3.1.6-1 项目架构概览（蓝色：存储结构，绿色：逻辑结构，橘色：前端结构）

### 3.1.8. 存储部分

本项目使用了多种组件——使用阿里云 OSS 存储房地产代币的基础资料和

房地产预览图；使用以太坊存储智能合约运行过程中产生的必要数据，如投资者持有的代币数量、房屋租赁信息、代币流动信息等；使用 MySQL 存储周期性任务的配置和统计信息，如投资者累计收益等；将 Redis、Celery 与 web3.py 配合，以便周期性监听区块链上的特定事件，从而实现对智能合约运行状态的监控。Redis、Celery 和 web3.py 的结合，将区块链上的数据合计汇总存入 MySQL，实现了去中心化与中心化两部分数据的互通，为项目的异步部分，如房地产代币发行和房地产租赁功能奠定了基础。

### 3.1.9. 逻辑部分

逻辑部分分为两个模块——智能合约部分实现了项目的核心逻辑，包含房地产代币的发行、销毁、转账、销售、房地产租赁、收益分红等功能，运行在以太坊上。Etherscan API 和 OpenSea API 提供部分投资者交易信息和房地产代币信息，数据来源于以太坊，但由中心化服务器提供。Django 部分同样运行在中心化服务器上，主要包含一些附加功能，如房地产代币信息录入、投资者收益统计等功能。Django REST framework 模块作为 Django 框架的补充，为前端提供了 Restful API。

### 3.1.10. 前端部分

本项目使用 Flutter 进行开发。UI 部分，实现了房地产浏览、租赁，交易查询等页面。逻辑部分，实现了与各个 API 的交互，其中，flutter\_web3 模块提供 flutter 与以太坊交互的底层功能，dio 模块提供 Flutter 与中心化服务器的交互功能。整体来看，实现了包括从阿里云 OSS 中请求房地产信息，从 Django 中心化服务器中请求历史收益信息，从 Etherscan API、OpenSea API 和智能合约中请求交易信息和投资者信息等功能。

## 3.2. 架构解读

本项目的存储部分和逻辑部分均被拆分为几乎独立的两个模块，这是由区块链的特性决定的。

### 3.2.1. 存储部分

本项目的核心功能基于以太坊，因此服从以太坊的运行机制。用户的任何操作，只要涉及对区块链上数据的修改，就需要支付 gas 费。一般来说，gas 费的收取与链上数据修改的程度有关——修改内容越多，需要支付的 gas 费越多；修改的数据越持久，需要支付的 gas 费越多。

因此，将房地产的描述信息、图片信息、投资者收益统计信息等存储在区块链上将是十分昂贵的。这迫使我们在设计智能合约之初，就需要将核心信息和周边信息区分开。智能合约将仅存储合约运行所必需的数据，其余数据依据功能，应该存储与 OSS 服务器，或者其他中心化服务器中。

房地产代币描述文件和房地产预览图属于频繁读取的静态文件资源，应该存储在单独的文件服务器中，如阿里云 OSS；收益统计信息需要结合智能合约运行状态定期维护和更新，因此存放在中心化的 MySQL 数据库中。

当用户使用应用时，前端根据实际需求，分别从静态资源服务器中获取房地产代币描述文件和房地产预览图，从 Django 服务器中获取用户收益信息等动态资源。如此“动静分离”的结构，可以大幅提升项目的整体性能。

由于需要定期统计投资者的收益信息和智能合约的运行状态信息，因此需要使用作业调度程序 Celery 实现对链上数据的定期监听。这里使用 Redis 充当消息队列，Celery beat 实现作业的周期性调度，web3.py 实现对以太坊的访问，Celery worker 实现作业的执行。

### 3.2.2. 逻辑部分

本项目的核心功能运行于以太坊，并且运行过程不能与链外交互。因此，本项目的两个逻辑部分也几乎独立运行。

Django 和 Django Restful framework 实现了对 MySQL 数据库的管理和访问功能，如获取某投资者某段时间内的收益信息，和异步的房地产注册功能。

智能合约作为核心信息的生产者和维护者，自然也会提供核心信息的 API。例如，获取某投资者持有的代币数量、获取某房地产的租赁日期、到期日期、租金、获取某投资者对某房地产的租金投票时，应当提供的合法的租金范围等。

Etherscan API 和 OpenSea API 是由第三方公司维护的 API。Etherscan API 主要提供智能合约的交易信息，例如某交易发生的时间、区块、执行结果、调用者、执行参数等。OpenSea API 主要提供代币的流动信息，如发行时间、转账金额、销售或拍卖信息等。

由于区块链的公开性，这些 Etherscan API 和 OpenSea API 提供的信息在以太坊的全节点上可查的，但我们依然不得不使用这些第三方 API 来获取这些数据。因为链上记录了来自全球的交易，如果持续监听链上的每一笔交易，并对每一笔交易建立索引，就需要消耗大量资源。本项目在运行过程中会产生许多交易，



因此使用自建的服务器完成这些琐碎而繁重的工作是不现实的。综上所述，这里使用 Etherscan API 和 OpenSea API，共两个来自第三方的接口，来完成部分信息的获取。

### 3.3. 技术选型

为了实现基于区块链的房地产交易平台，需要对关键技术选型，这些技术主要包括区块链平台、智能合约框架、中心化部分的后端框架和服务器、前端框架等。

#### 3.3.1. 区块链平台

我们的目的是开发一款去中心化的应用程序，因此区块链的市值不是我们关心的主要内容。我们更关心的是，在加密货币的市值背后，区块链本身是否可信、安全，以及区块链可以如何赋能我们的项目，例如，项目的稳定性、开发的便捷性等。

项目筹备初期，我们比较了三款常见的区块链平台——比特币、以太坊、EOS。

比特币，其使用的脚本语言虽然可以支持许多操作，但是并不支持循环控制和条件控制语句。这种设计其实是有意为之的，因为这样可以限制脚本的复杂度，从而避免比特币节点被攻击。然而，这种设计使得该脚本语言不是图灵完备的，意味着我们无法在比特币网络上完成复杂的交易。

以太坊的代码运行在以太坊虚拟机中，可以支持复杂的流程控制和条件判断语句，这意味着该语言是图灵完备的。我们可以在以太坊上开发出复杂的交易流程，但执行这些复杂的交易流程需要支付一定的 gas 费。因为 gas 费的存在，我们需要使交易流程尽可能简单，从而尽量少使用 gas 费。并且，以太坊拥有较为丰富的开发资料，例如 OpenZeppelin 为智能合约的开发提供了底层框架，使得开发者在没有丰富的开发经验的情况下也可以开发出稳定安全的智能合约。

EOS 是一个区块链平台，为去中心化应用的运行提供了基础服务。深入了解 EOS 后，我们发现 EOS 与比特币或以太坊不同——EOS 是联盟链，仅部分超级节点有记账权；比特币和以太坊是公有链，任何人都可以加入，成为记账者。EOS 中，仅有 21 个全节点拥有记账权，即——区块链的共识是 21 个有记账权的节点的共识，并不是所有用户的共识。这样的机制可能引起垄断，使区块链朝向中心化的趋势发展。

综上所述，比特币不具备开发本项目的基础；EOS 作为联盟链，其本身的可

信度存疑；基于以太坊的交易虽然需要支付 gas 费，但它提供了可信的环境和较为完备的开发资源，因此我们选择以太坊，作为本项目的底层区块链平台。

### 3.3.2. 智能合约协议

为了更好地融入以太坊生态，我们选择基于现成的代币协议进行开发。目前，主要的代币协议有 ERC-20、ERC-721、ERC-1155。这三种协议分别代表不同的应用场景，我们对此进行了了解，认为 ERC-1155 协议更符合本项目的使用场景。

ERC-20 是一种同质化代币，这意味着基于同一协议的不同代币之间没有区别。生活中，如人民币就是一种同质化代币，因为人民币可以等额互换，不同的人民币之间没有区别[3]。

ERC-721 是一种非同质化代币，这意味着每一个代币都是独一无二的。一般来说，他们拥有不同的价格，也不能相互兑换。ERC-721 代币常用于艺术品领域，例如书法艺术品[4]。

ERC-1155 结合了 ERC-20 与 ERC-721。在同一个 ERC-1155 协议中，可以发行多种代币，这些代币既可以是类似 ERC-20 的同质化代币，也可以是类似 ERC-721 的非同质化代币。ERC-1155 经常用于游戏领域。例如某游戏可以发行 1000 枚斧子、100 把剑和 1 枚水晶。斧子之间可以互换，是同质化的。剑之间也可以互换，也是同质化的。水晶只有一枚，是非同质化的[5]。

本项目中，我们尝试对不同的房地产发行不同的代币，同一房地产的代币是同质化的，不同房地产代币是非同质化的。

若我们使用 ERC-20 协议，用户每注册一处房地产，就需要部署一个新的 ERC-20 协议。部署协议将花费大量 gas 费，因此我们并不把 ERC-20 协议放在首选地位。若我们使用 ERC-721 协议，虽然可以为不同的房地产发行不同的代币，但是每种代币是不可分的，这不能实现本项目的初衷——让投资者以较低投入进入房地产市场。

ERC-1155 协议正好适合本项目，因为 ERC-1155 协议可以发行多种代币，不同代币之间不能互换，但相同代币之间可以互换。并且，每次发行代币不需要部署新的合约，可以有效降低 gas 费的消耗。

综上所述，我们使用 ERC-1155 作为本项目的核心协议。

### 3.3.3. 中心化部分的后端框架

毫无疑问，我们的首选框架是 Django。主要原因是我们有着丰富的 Python

开发经验，而 Django 是使用 Python 开发的后端框架。除此之外，Django 有许多优点，这些优点可以帮助我们更高效地完成后端开发。

社区：Django 有一个庞大的、活跃的社区。我们的任何问题都可以在社区中快速找到答案。

久经考验：Django 自 2005 年就被广泛使用，有许多大型项目都基于 Django 开发，并且使用了许多年，例如豆瓣。

良好的扩展性：Django 拥有庞大的生态，有许多第三方包可供选择，例如我们接下来介绍的 Django Restful framework。

#### 3.3.4. Django Restful framework

Django 是一款前后端不分离的网站开发框架。在本世纪初，互联网刚刚兴起时，前后端不分离是主流的开发方式。近十年，随着移动互联的发展，越来越多的网站需要针对不同尺寸的设备开发不同的前端。因此，前后端分离的模式逐渐流行。在这种模式下，前端与后端解耦，后端只负责网站的逻辑部分，前端只负责对后端数据的渲染。本项目为了实现更清晰的前后端逻辑，也为了适应不同平台的前端框架，选择前后端分离的结构。我们使用 Django Restful framework 开发后端 API，这样可以使 Django 更好地支持前后端分离，同时也可以为前端提供标准的接口。

#### 3.3.5. 中心化部分的数据库

中心化部分主要处理和存储收益统计信息和智能合约运行的状态信息，因此我们选择关系型数据库。这里我们使用 MySQL 作为中心化部分的数据库，不只是因为我们对于 MySQL 拥有更多的开发经验，还因为 Django 对 MySQL 拥有良好的兼容性。综上所述，我们认为 MySQL 凭借其强大的性能和出色的兼容性可以胜任这一角色。

#### 3.3.6. 对象存储服务

我们将房地产描述信息和预览图这类静态文件存储在独立的静态资源服务器中，将其他数据存储在 MySQL 或以太坊中。为此，我们需要选择一款合适的存储服务。阿里云、腾讯云、华为云等云服务商均有提供对象存储服务，其工作原理、性能和收费方式大致相同。我们最终选择了阿里云 OSS 对象存储服务，因为我们更熟悉阿里云的产品，未来也计划将 Flutter 前端、Django 服务等部署于阿里云服务器。我们尽量在同一个平台部署项目，这样可以降低管理成本。

### 3.3.7. 前端框架

我们使用 Flutter 开发网站的前端部分，因为 Flutter 作为一款前端框架，可以帮助开发者高效开发跨平台的前端。不论是 Android，还是 IOS，抑或是 Windows、Web，Flutter 可以实现一次编码，多平台使用。并且 Flutter 拥有高效的性能和完备的 UI 组件，可以帮助我们便捷地开发出项目前端。

## 4. 项目实施

### 4.1. 构建智能合约

OpenZeppelin 提供了完善的 ERC-1155 协议的实现，其已经实现了完整的代币发行、转账和销毁等功能。我们继承了 OpenZeppelin 的 ERC-1155 合约，定制了合约的部分参数，并实现了租赁相关的功能。

我们首先定义了租赁信息结构体，用来存储房地产的租赁状态，如房地产 id、租约起止时间、租户账户、租金等内容。

之后我们实现了租金投票、分红、代币流动时的租金刷新等功能。

### 4.2. 实现异步的注册房地产功能

由于区块链的特性，用户的每一笔交易都需要矿工进行打包，并且由于区块链分布式的特性，用户的每一笔交易被打包的时间可能不确定。如果用户提供足够高的 gas 费，就可以吸引矿工优先对该交易打包；但如果用户提供的 gas 费过低，该交易可能永远不会被打包。由于交易被确认的时间不确定，所以本项目的注册房地产、租赁房地产、房地产代币转账等功能需要异步执行。即，用户发出交易后，交易结果不会立刻收到，而是在交易被确认后才能收到。异步的特性提高了某些需要 Django 和智能合约共同完成的功能的实现难度。本小节将着重阐述这部分功能的实现方法。

这里以注册资产功能为例。

注册资产操作需要同时向智能合约和中心化部分的 Django 服务提交请求。智能合约将帮助投资者发行房地产代币，中心化部分的 Django 服务将帮助用户上传房地产描述信息和预览图。难点在于，中心化部分的 Django 服务不能及时获取智能合约的运行状态。要想克服此问题，目前有两种实现思路。

方法一：Django 服务接收到来自用户的房地产注册请求后，使用轮询的方式检查 MySQL 数据库，获取智能合约的运行状态。一旦检测到智能合约完成了房地产代币发行，便上传房地产描述信息和预览图到阿里云 OSS 服务器。

方法二：Django 接收到来自用户的房地产注册请求后，暂存房地产描述信息和预览图到 MySQL 服务器和静态文件服务器，然后立刻返回结果。此时，Celery 的周期性任务将持续检测智能合约的运行状态。周期性任务一旦检测到智能合约发行了房地产代币，便回调 Django 上传房地产描述信息和预览图的函数。

上述介绍了两种克服中心化部分的 Django 服务不能及时获取智能合约运行状态的方法。方法一的逻辑简单，易于实现，但会消耗大量服务器资源，也会把用户阻塞在提交注册房地产页面，直到 Django 完成上传房地产描述信息和预览图。方法二属于异步的解决方案，可以节约服务器资源，也可以提升用户体验，但实现过程较为复杂。

出于节约服务器资源，提升用户使用的考虑，我们最终决定使用方案二。图 4.3.7-1 是方案二的运行逻辑图。

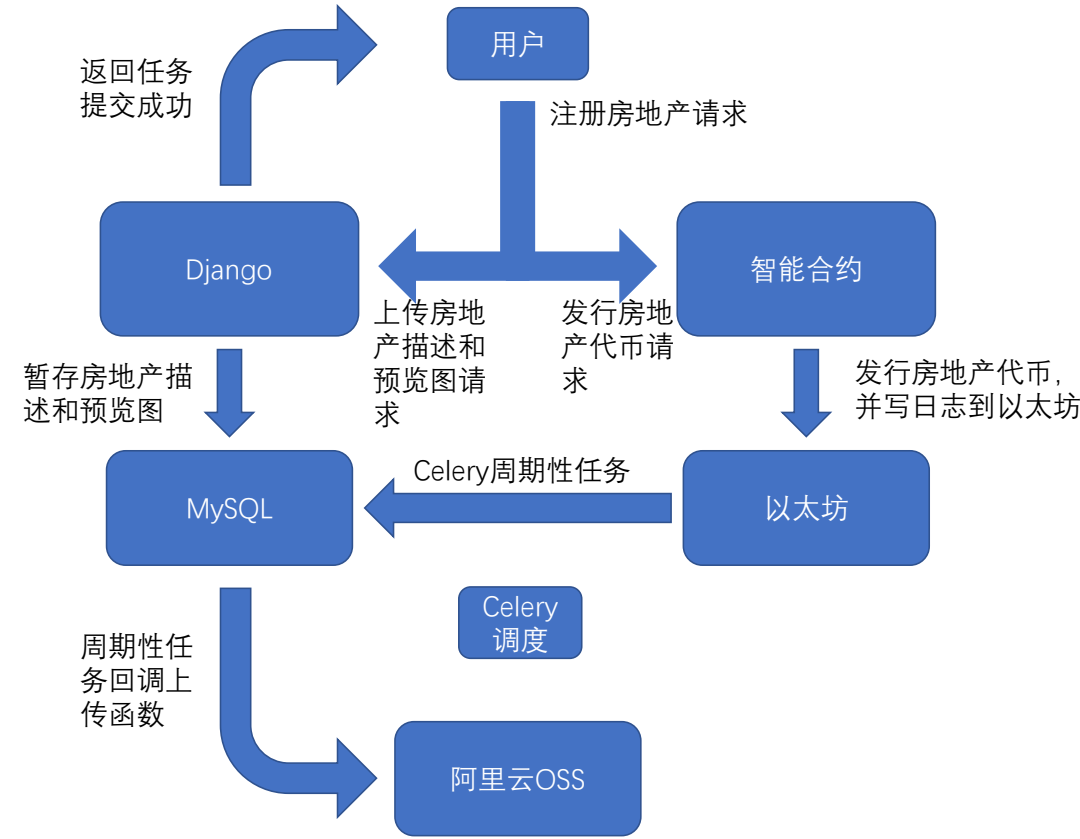


图 3.3.7-1 异步的注册房地产流程图

### 4.3. 实现监听智能合约运行状态

监听智能合约运行状态主要是通过监听链上事件的方式实现的。我们主要使用 `django-ethereum-events` 实现了周期性任务；使用 Celery 实现了调度。

周期性任务将连接到以太坊全节点，检查最近区块中的发生的事件。若发现

指定的合约产生了指定的事件，随即获取事件信息，如分红事件的发生日期、受益人和房地产 id 等信息，然后存入数据库。

#### 4. 4. 实现前端

前端的实现没有特殊性，这里不做详细介绍。

## 结论

本文首先对房地产市场的现状进行了分析，认为房地产市场出现了流动性困难、交易手续繁杂和存在欺诈现象等问题。继而，我们提出了使用区块链技术来缓解以上问题的思想，并对其进行了实现。本项目基于区块链，主要实现了常见的房地产投资行为，如房地产代币的发行、转账、销售和房地产租赁、收益分红等功能。我们认为这些功能可以在一定程度上缓解房地产市场中遇到的问题，但房地产去中心化治理道阻且长，不仅需要技术支持，更需要相关法律法规的完善。虽然目前相关技术和法律仍不完善，但我们相信，区块链将在房地产市场发挥越来越重要的作用。

## 参考文献

- [1] 清华大学五道口金融学院. 商业房地产的代币化和流动性[EB/OL]. [2019-08-16]. <https://page.om.qq.com/page/OILHgu2-pVy47F6fB698Viuw0>.
- [2] JOE LIEBKIND. How Blockchain Technology is Changing Real Estate[EB/OL]. [2022-3-22]. <https://www.investopedia.com/news/how-blockchain-technology-changing-real-estate/>.
- [3] wackerow. ERC-20 TOKEN STANDARD[EB/OL]. [2021-12-4]. <https://ethereum.org/en/developers/docs/standards/tokens/erc-20/>.
- [4] corwintines. ERC-721 NON-FUNGIBLE TOKEN STANDARD[EB/OL]. [2022-3-10]. <https://ethereum.org/en/developers/docs/standards/tokens/erc-721/>.
- [5] corwintines. ERC-1155 MULTI-TOKEN STANDARD[EB/OL]. [2022-3-10]. <https://ethereum.org/en/developers/docs/standards/tokens/erc-721/>.

## 致谢

作者在设计（论文）期间都是在王莉教授全面、具体指导下完成进行的。王莉老师渊博的学识、敏锐的思维、民主而严谨的作风使学生受益非浅，并终生难忘。